A Fuzzy Relational Clustering Algorithm with $q$-weighted Medoids

Ying GAO$^{1,2,*}$, Hong Qi$^{1,2}$, Dayou Liu$^{1,2}$, Jiafei Li$^{1,2}$, Lina Li$^{1,2}$

$^{1}$College of Computer Science and Technology, Jilin University, Changchun 130012, China
$^{2}$Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China

Abstract

Medoids-based fuzzy relational clustering generates clusters of objects based on relational data, which records pairwise similarity or dissimilarities among objects. Compared with single-medoid based approaches, multiple-weighted medoids has shown superior performance in clustering. In this paper, we present a new version of fuzzy relational clustering in this family called fuzzy clustering with $q$ weighted medoids (FQMdd). Based on the objective function of FQMdd, the update of memberships, prototypes of each cluster and weights are given. Compared with two existing fuzzy relational clustering approaches fuzzy c-medoids (FCMdd) and fuzzy clustering with multi-medoids (FMMdd), FQMdd can represent rich cluster-based information by multiple-weighted objects, and strengthen contribution of core objects in clusters as well. Experiments on synthetic and real-world datasets show that FQMdd mostly has higher clustering accuracy than FCMdd and FMMdd.

Keywords: Clustering Analysis; Fuzzy Clustering; Relational Data; Weighted Medoids

1 Introduction

Clustering methods organize a set of items into clusters such that items within a given cluster have a high degree of similarity, whereas those of different clusters have a high degree of dissimilarity. These methods have been widely applied in many fields such as pattern recognition, data mining, computer vision, computational biology, etc [1, 2].

There are usually two kinds of representations of the data sets upon which clustering algorithm can be based [3]. One is vector representation, where each object is represented as a vector in some feature space, and each item of vector can be numerical or categorical. This kind of data is also called feature data. Many traditional clustering algorithms such as fuzzy c-mean

*The work is supported by National Natural Science Foundation of China under Grant No. 61202308, 61133011, 61373053. Key Programs for Science and Technology Development of Jilin Province of China under Grant No. 20130206046GX.

*Corresponding author.
Email address: gying@jlu.edu.cn (Ying GAO).
Partitioning around medoids (PAM) [5] are based on this kind of data. The other kind of data representation is relational representation, where the available datasets exist in the form of pairwise relationships. The relationships describe the similarity or dissimilarity between each pair of objects. Normally, relational data can be derived from feature data through proper proximity or distance measures. However, it is difficult to derive feature data from relational data. So, many clustering algorithms are developed for relational data, such as relational fuzzy \( c \)-means clustering (RFCM) [6], the fuzzy \( c \)-medoids (FCMdd) [7], clustering and aggregation of relational data (CARD) [8], fuzzy relational clustering with multi-medoids (FMMdd) [9], partitioning hard clustering algorithms based on multiple matrices (MRDCA) [10], etc.

Partitioning method is one of the most popular clustering techniques. Partitioning method seeks to obtain a single partition of the input data into a fixed number of clusters, and often looks for a partition that optimizes (usually locally) an objective function. In existing partition-focused clustering algorithms, medoid-based ones can handle any form of data without additional computations, and can provide additional information on the relative importance or representativeness of objects in each cluster, which is very useful for better understanding and description of each cluster produced.

Some partitioning algorithms such as PAM [5], FCMdd [7], produce clusters where each of them is represented by one representative object or medoid. However, in some cases, it may not be sufficient enough to use only one object to represent the whole cluster. In a recent work, a version of fuzzy relational clustering called fuzzy clustering with multi-medoids (FMMdd) was presented [9], which used multiple weighted objects to represent a cluster, and produced good quality of clusters with rich cluster-based information. In FMMdd, a matrix \( V_{K \times N} \) was used to record prototype weights of \( N \) objects with respect to all the \( K \) clusters, and all objects were engaged in describing clusters information. However, it is easy to understand that when an object does not belong to a cluster, it should not participate in describing the cluster.

In this paper, based on FMMdd, we present a new version of fuzzy relational clustering algorithm called fuzzy clustering with \( q \) weighted medoids (FQMdd), in which we use \( q \) weighted objects to represent the prototype of each cluster, and the prototypes and weights are computed based on the representing matrix \( V_{K \times N} \) in FMMdd. Compared with FMMdd, the purpose of FQMdd is selecting \( q \) core objects as prototype of each cluster, and increasing contributions of core objects to represent clusters and eliminating contributions of other objects as well.

In the next section, we introduce related work of FCMdd and FMMdd. After that, we give our objective function, solution, and algorithm in Section 3. To illustrate and verify our algorithm, several experiments are set up in Section 4. Finally, we conclude this study in Section 5.

## 2 Related Work

Let \( X = \{x_i | i = 1, 2, ..., N\} \) be a set of \( N \) objects. Let \( R \) be the dissimilarity matrix, where \( r_{ij} \geq 0 \in R \) records the relationship or dissimilarity between two objects \( x_i \) and \( x_j \). Supposing the number of clusters is \( K \). Matrix \( U_{K \times N} \) is membership matrix, where \( u_{ci} \in [0, 1] \) denotes the fuzzy membership of \( x_i \) for cluster \( c \).
2.1 FCMdd

FCMdd [7] is one of the most popular fuzzy relational clustering algorithms. In FCMdd, the prototype of each cluster is represented by one object. Let \( V = \{v_1, v_2, ..., v_K\} \) represent prototypes of \( K \) clusters, where \( v_c \in X(c = 1, 2, ..., K) \). FCMdd minimizes:

\[
J_{FCMdd} = \sum_{i=1}^{N} \sum_{c=1}^{K} u_{ci}^m r(x_i, v_c)
\]

where parameter \( m \in [1, \infty) \) is the “fuzzifier”.

2.2 FMMdd

FMMdd [9] is a recent work of fuzzy relational clustering with multi-medoids. In FMMdd, prototypes of clusters are represented by a matrix \( V_{K \times N} \), which records the prototype weights of \( N \) objects with respect to \( K \) clusters, and subjects to \( v_{cj} \geq 0 \) for each \( c \) and \( j \), and \( \sum_{j=1}^{N} v_{cj} = 1 \) for each \( c \).

The objective of FMMdd is to minimize the following criterion:

\[
J_{FMMdd} = \sum_{c=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{N} u_{ci}^m v_{cj}^n r_{ij}
\]

where parameter \( m \) controls the fuzziness of memberships, \( n \) controls the smoothness of prototype weights, and \( r_{ij} \) is abbreviation of \( r(x_i, x_j) \).

3 The Proposed Approach: FQMdd

In this section, we present a new version of fuzzy relational clustering with multiple weighted medoids, in which neither one object is used to represent the prototype of each cluster like in FCMdd, nor all weighted objects are used to represent a prototype like in FMMdd, but a subset of fixed cardinality \( 1 \leq q \ll N \) of objects \( X \) are used. This algorithm is called FQMdd. These \( q \) objects constitute core of each cluster, and they collaborate to describe information of the cluster.

3.1 Objective function

Let \( G_c(c = 1, 2, ..., K) \) be the prototype of cluster \( c \), which include \( q \) objects, i.e. \( G_c = \{g_{c1}, g_{c2}, \ldots, g_{cq}\} \), \( W_c = (w_{c1}, w_{c2}, \ldots, w_{cq}) \) be weights of objects in \( G_c \), subjects to \( \sum_{j=1}^{q} w_{cj} = 1 \). The objective of FQMdd is to minimize

\[
J_{FQMdd} = \sum_{c=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{q} u_{ci}^m w_{cj} r(x_i, g_{cj})
\]
3.2 Solutions

We need to solve the above minimization problem by finding the optimal $U$, $G$ and $W$. Like FCMdd and FMMdd, FQMdd also starts with randomly initialization, and iteratively updates $U$, $G$ and $W$ respectively, until objective function reaches a stability value or the maximum number of iteration times is reached.

(1) Update of $U$.

When $G$ and $W$ are fixed, the update equation of $U$ is:

$$u_{ci} = \frac{\left( \sum_{j=1}^{q} w_{cj} r(x_i, g_{cj}) \right)^{-1/(m-1)}}{\sum_{f=1}^{K} \left( \sum_{j=1}^{q} w_{fj} r(x_i, g_{fj}) \right)^{-1/(m-1)}}$$

(4)

The proof is the same as that in the classical fuzzy $K$-means algorithm [4].

(2) Update of $G$ and $W$.

We know that, in FMMdd, matrix $V_{K\times N}$ records the weights of objects representing clusters. The larger $v_{cj}$ is, the better that $x_j$ represents cluster $c$. The update equation of $V$ is Eq. (5), which is derived by the method of Lagrange multipliers in Ref [9].

$$v_{cj} = \frac{\left( \sum_{i=1}^{N} u_{ci}^{m} r_{ij} \right)^{-1/(n-1)}}{\sum_{h=1}^{N} \left( \sum_{i=1}^{N} u_{ci}^{m} r_{ih} \right)^{-1/(n-1)}}$$

(5)

In FQMdd, we compute $G$ and $W$ according to matrix $V_{K\times N}$ in FMMdd. The prototype of each cluster in FQMdd is composed of $q$ objects with maximum weights in each row of $V$. The weights of objects in prototype are the relative values in $V$, but they should be normalization.

The algorithm of update of $G$ and $W$ is described as algorithm 1.

**Algorithm 1: Update Prototypes and Weights**

**Input:** $V_{K\times N}$.

**Output:** $G_c = (g_{c1}, g_{c2}, \ldots, g_{cq}), W_c = (w_{c1}, w_{c2}, \ldots, w_{cq}) (c = 1, 2, K)$.

for $c = 1$ to $K$

\{  
  for $j = 1$ to $q$
    
  \{$g_{cj} = \arg\max_{j=1,\ldots,N}(V(c, j))$;  
  \$w_{cj} = \max_{j=1,\ldots,N}(V(c, j)); \$  
  \$V(c, g_{cj}) = 0; \$
  \}  
  \$\text{SumW}=\text{sum}(w_{c1}, w_{c2}, \ldots, w_{cq});\$
  \$\text{for } j = 1 \text{ to } q$
  \$w_{cj} = w_{cj}/\text{SumW};\$
  \}

3.3 Algorithm of FQMdd

The relational clustering algorithm FQMdd is summarized as algorithm 2.
Algorithm 2: Fuzzy relational clustering with q weighted medoids

Input: \( R_{N \times N} \) (dissimilarity matrix), \( K \) (the number of clusters), \( q \) (the cardinality of prototype), parameter \( 1 < m, n < +\infty, 0 < \varepsilon \ll 1, T \) (an iteration limit).

Output: membership matrix \( U_{K \times N} \), prototypes of clusters \( G_c \) and their weight \( W_c \) (\( c = 1, 2, \ldots, K \)).

(1) Initialization.
Set \( t = 0 \);
Randomly select \( K \) distinct prototypes \( G_c \) (\( c = 1, 2, \ldots, K \)), and each prototype includes \( q \) objects, i.e. \( G_c^{(0)} = (g_{c1}^{(0)}, g_{c2}^{(0)}, \ldots, g_{cq}^{(0)}) \) (\( c = 1, \ldots, K \));
Set weights \( W_c^{(0)} = (1, 1, \ldots, 1) \) (\( c = 1, \ldots, K \));
Compute \( U^{(0)} \) according to Eq. (4);
Compute objective \( J^{(0)} \) according to Eq. (3).

(2) Update of \( G \), \( W \) and \( U \).
Set \( t = t + 1 \);
Fix \( U^{(t-1)} \), compute representing matrix \( V \) according to Eq. (5).
Update prototypes and weights based on \( V \) according to algorithm 1, and get \( G^{(t)}, W^{(t)} \);
Fix \( G^{(t)}, W^{(t)} \), compute \( U^{(t)} \) according to Eq. (4).

(3) Stopping criterion.
Compute \( J^{(t)} \) according to Eq. (3). If \( |J^{(t)} - J^{(t-1)}| < \varepsilon \) or \( t > T \), STOP; otherwise goto (2).

4 Experimental Results

In this section, several experiments are set up to evaluate the performance of FQMdd. We compare our FQMdd with FMMdd and FCMdd on both synthetic and real-world datasets.

4.1 Synthetic dataset

First, we compare the results of FQMdd with FCMdd and FMMdd to show that \( q \)-weighted objects are able to represent cluster better than a single object or all \( N \) objects. The test dataset is originally used in [3]. The distribution of synthetic dataset is shown in Fig. 1, and the corresponding coordinates are included in Table 1. It can be seen from Fig. 1 that object \( x_1 \) to \( x_4 \) form one cluster, and \( x_5 \) to \( x_{10} \) form the other cluster. Object \( x_{11} \) is located near the middle of two clusters but slightly biased towards the second cluster. The dissimilarity matrix is calculated using Euclidean distance and is scaled as \( r_{ij} / \max_{ij} \{d_{ij}\} \). We set \( K = 2, \varepsilon = 1.0 \times 10^{-5} \) for these three approaches, and \( m = 1.9 \) for FCMdd. \( m = 1.9, n = 1.5 \) are used for FMMdd. \( m = 1.9, n = 1.5, q = 4 \) are used for FQMdd. The results of FCMdd, FMMdd, FQMdd are given in Table 1.

Table 1 shows that, partitions produced by these methods are similar, which indicates that \( x_1 \) to \( x_4 \) are in the first cluster \( C_1 \), and \( x_5 \) to \( x_{10} \) are in the second cluster \( C_2 \). For FCMdd, \( x_4 \) and \( x_7 \) are medoids (representative objects) for \( C_1 \) and \( C_2 \), respectively, and \( x_{11} \) is assigned to \( C_1 \), since \( x_{11} \) is closer to \( x_4 \) than \( x_7 \), this misleading is due to its single-medoid representation. For FMMdd, all objects are engaged in representing each cluster, but their weights are different. \( x_1 \) to \( x_4 \) have larger weights for representing \( C_1 \) than \( C_2 \), and \( x_5 \) to \( x_{10} \) have larger weights for representing
Table 1: Results of FCMdd, FMMdd and FQMdd on synthetic dataset

<table>
<thead>
<tr>
<th>i</th>
<th>Coordinates</th>
<th>FCMdd ((m = 1.9))</th>
<th>FMMdd ((m = 1.9, n = 1.5))</th>
<th>FQMdd ((m = 1.9, n = 1.5, q = 4))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>x y u1i u2i Medoids u1i u2i v1i v2i</td>
<td>u1i u2i v1i v2i</td>
<td>u1i u2i v1i v2i</td>
<td>u1i u2i v1i v2i</td>
</tr>
<tr>
<td>1</td>
<td>-1.2 1.0 0.8681 0.1319</td>
<td>0.8989 0.1011 0.2167 0.0022</td>
<td>0.9166 0.0834 0.2298</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>-1.0 1.2 0.8974 0.1026</td>
<td>0.952 0.1048 0.2339 0.0027</td>
<td>0.9218 0.0882 0.2484</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>-1.0 0.8 0.8972 0.1028</td>
<td>0.952 0.1048 0.2339 0.0027</td>
<td>0.9115 0.0885 0.2482</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>-0.8 1.0 1.0000 0.0000</td>
<td>0.905 0.1095 0.2573 0.0034</td>
<td>0.9502 0.0948 0.2737</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.8 1.0 0.8186 0.9184</td>
<td>0.976 0.9220 0.0074 0.1357</td>
<td>0.9173 0.9287 0.1799</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1.0 1.2 0.8880 0.9120</td>
<td>0.972 0.9279 0.0059 0.1227</td>
<td>0.9094 0.9106 0.1277</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.98 0.98 0.9000 1.0000</td>
<td>0.933 0.9667 0.0061 0.2441</td>
<td>0.9277 0.9723 0.1833</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1.0 1.0 0.8207 0.9701</td>
<td>0.933 0.9667 0.0058 0.2380</td>
<td>0.9323 0.9677 0.2322</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1.0 0.8 0.719 0.9281</td>
<td>0.972 0.9280 0.0059 0.1227</td>
<td>0.9652 0.9348 0.1639</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.2 1.0 0.8796 0.9284</td>
<td>0.967 0.9325 0.0049 0.1134</td>
<td>0.9822 0.9178 0.1799</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.05 1.0 0.5250 0.4750</td>
<td>0.4421 0.5579 0.023 0.124</td>
<td>0.4626 0.5374 0.127</td>
<td></td>
</tr>
</tbody>
</table>

C2 than C1. The weights of x11 for representing C1 and C2 are both small. The membership of x11 for C2 is larger than that for C1. It is reasonable that FMMdd assigns x11 to C2, but it is unreasonable that objects that do not belong to one cluster have contributions to represent the cluster, even if their weights are small. For FQMdd, we select q=4 objects with maximum weights as core objects to represent each cluster, and distribute weights of other objects to these q objects. x1 to x4 are selected to represent C1, and x5, x7, x8, x9 are selected to represent C2. The memberships and weights of core objects are larger than that of FCMdd and FMMdd, which are more reasonable.

4.2 Real-world datasets

In this experiment, we test performances of FQMdd, FMMdd and FCMdd on five real-world UCI datasets. These datasets are summarized in Table 2 and can be found in [http://archive.ics.uci.edu/ml/](http://archive.ics.uci.edu/ml/).

Table 2: Summary of real-world UCI datasets

<table>
<thead>
<tr>
<th>dataset</th>
<th>N (number of objects)</th>
<th>P (number of attributes)</th>
<th>K (number of clusters)</th>
</tr>
</thead>
<tbody>
<tr>
<td>iris</td>
<td>150</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>wine</td>
<td>178</td>
<td>13</td>
<td>3</td>
</tr>
<tr>
<td>glass</td>
<td>214</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>wdbc</td>
<td>569</td>
<td>30</td>
<td>2</td>
</tr>
<tr>
<td>seeds</td>
<td>210</td>
<td>7</td>
<td>3</td>
</tr>
</tbody>
</table>

To evaluation clustering results furnished by clustering algorithms, we consider two external evaluation metrics Corrected Rand (CR) index [11] and F-Measure [12]. These evaluation metrics assess the degree of agreement (similarity) between an a priori partition and a partition furnished by the clustering algorithm. Let \(P = \{P_1, P_2, \ldots, P_m\}\) be the a priori partition into \(m\) classes and \(Q = \{Q_1, Q_2, \ldots, Q_K\}\) be the hard partition into \(K\) clusters given by a clustering algorithm.
We test each algorithm using different parameters on each dataset. For iris dataset, Fig. 2 is the CR index and F-measure results of FCMdd when parameter $m$ changes from 1.5 to 2.5. We can see from Fig. 2 that the CR index and F-measure is 0.74453 and 0.89828 respectively, which are not sensitive to parameter $m$. Fig. 3 and Fig. 4 are the CR index and F-measure results of FMMdd when parameter $m$ and $n$ change from 1.0 to 2.0. We can see from Fig. 3 and Fig. 4 that, the results of FMMdd are sensitive to parameters $m$ and $n$. The maximal CR index of FMMdd is 0.8176, and the maximal F-measure is 0.93331.

To compare FMMdd and FQMd, we set $m = 1.7, n = 1.1$, which are better parameters according to Fig. 3 and Fig. 4, and test FQMdd for different $q$. The CR index and F-measure results of FQMdd are in Fig. 5 when $q$ changes from 7 to 20. The maximal CR index of FQMdd is 0.85084, and the maximal F-measure is 0.94667.
For other datasets, we select parameters and compare the maximal \( CR \) index and \( F\)-measure by the same method as for iris dataset. Table 3 lists the parameters we selected, and Table 4 lists the \( CR \) index and \( F\)-measure of FCMdd, FMMdd and FQMdd based on the parameters.

We can see from Table 4 that, for most of the five datasets, FQMdd can obtain better clustering results than FCMdd and FMMdd. This may because multiple weighted core objects representative for each cluster can capture complicated cluster structures more accurately than one object or all objects.

5 Conclusions

In this paper, we work on the medoids-based fuzzy relational clustering methods. Based on a recent work of fuzzy clustering with multi-medoids (FMMdd), we present a new version of fuzzy relational clustering called fuzzy clustering with \( q \)-weighted medoids (FQMdd). Compared with FMMdd, FQMdd selects fixed number of core objects to represent each cluster, strengthens contribution of core objects by eliminating other objects in prototype of each cluster, and learns the weights of core objects as well. FQMdd can capture complicated cluster structures more accurately. Experiments on synthetic and real-world datasets have shown the effectiveness of FQMdd.

References